

## CLAIMS

1. A method for routing data in wireless ad-hoc networks comprising the steps of:  
providing a proactive component; configured to route messages utilizing an intra  
zone routing protocol and

5

providing a reactive component; configured to route messages utilizing a inter  
zone routing protocol and

10

providing a bordercast tree, configured to bordercast to a plurality of border  
nodes; and

providing at least one query packet comprising data, wherein nodes receiving one  
or more query packets are configured to provide at least one query response or  
discard query packet;

15

wherein the method follows the following additional steps:

- i. a first route query is initiated by a first node or a source node and has one  
destination node;

20

- ii. if there is a path to a destination node in an outbound tree as computed by the  
proactive component, then that path is the desired path and the protocol  
terminates, otherwise;

iii. the source node checks if its bordercast tree is empty:

- a. if the bordercast tree is empty go to step viii;
- b. if the bordercast tree is not empty go to step iv;

5

iv. the bordercast tree is stored in the query packet, and is forwarded along the bordercast tree, and at least one intermediate nodes of the bordercast tree (non-border nodes), forward the query packet until it reaches a border node, wherein a plurality of processing steps occur culminating in the sending of a bordercast:

10

- a. after sending the bordercast, there is a pause for a predetermined period of time equal to ENHANCEMENT\_INTERVAL, during which the source node awaits either a query response or one or more enhancement messages;

15

v. if a query response to the route query is received, then the route query step is termed complete and the computed route is returned to the first node;

20

vi. if a query response is not received, then the source node checks if an enhancement message has been received, the ENHANCEMENT\_INTERVAL having passed since the initiation of the bordercast; if one or more query enhancement messages were received during the ENHANCEMENT\_INTERVAL, then one or more alternate destination nodes suggested in the query enhancement message, or messages, are utilized to create an enhanced route query with a alternative set

of destinations, wherein other nodes have reported that the alternative destination nodes have routes to the destination node; the new modified query is processed like the original route query; go to step ii;

5       vii.       if the bordercasting did not result in any enhancement of the route query or in a route, the bordercast tree is presumed incapable of reaching nodes that can enhance the query; this state is also reached from step ii when the bordercast tree is empty; in this situation a two-way tree is used to send a request to enhance the query, the source node and the border nodes forward this  
10       Query Enhancement Request using the two-way tree just as they would forward a regular query, except that the two-way tree is used for bordercasting, instead of the bordercast tree; the objective here is to try to discover nodes, which know of paths to the destination node;

15       viii.       after waiting for ENHANCEMENT\_INTERVAL, the source node checks to see if any responses to the query enhancement request using the two way tree exist, if one or more query enhancement responses are received during the ENHANCEMENT\_INTERVAL, the resulting one or more destinations suggested in the query enhancement response can be queried for routes to the desired  
20       destination, if there is a path to a desired destination node in an outbound tree as computed by the proactive component then; that path is the desired path; and the protocol terminates;

ix. if there are any responses to the query enhancement request but there is not a path to the desired destination in an outbound tree as computed by the proactive component then go to step **iii**;

5 x. if no enhancement message was received then the destination is assumed to be unreachable and the protocol terminates.

2. The method for routing data in wireless ad-hoc networks as set forth in claim 1, wherein in step **iv** the plurality of processing steps include:

- 10 a. receiving incoming query from step **iv** above;
- b. applying at least one query control mechanism to the query;
- c. using the criteria from the at least one Query Control Mechanism to drop unnecessary query threads,
- i. if all threads are dropped, then go to step **viii** of claim 1;
- 15 d. determining if a path is known to at least one destination and
- i. if a path is known notify the sender of the path and route discovery is complete;
- ii. if a path is not known;
- e. checking if a path is known from other nodes to at least one destination
- 20 node:
- i. if yes inform sender of alternative destinations;
- ii. if not goto step f;
- f. bordercasting query

g. returning to step **iv a** of claim **1**.

3. The method as set forth in claim **2** wherein the at least one query control mechanism of step **b** includes a Query Detection (QD) and an Early Termination (ET) mechanism; these mechanisms are configured to extract a query identifier  
5 and match it with recently cached query identifiers seen by the node; if the query identifier has been seen before, then the query can be dropped.

4. The method as set forth in claim **2** wherein the notification of a path, to the  
10 sender, of step **d i** is accomplished by:

a. initiating a response to the query that contains the computed path, only the border nodes, (also referred to as center nodes), traversed by the route query packet are recorded; the response is sent along a path that traverses the same center nodes; this is possible because each center node has a path  
15 to the previous center node, thus the response is forwarded from one center node to another center node until it reaches the source node, that initiated the query.

5. The method as set forth in claim **2** wherein the checking if a path is known from  
20 other nodes to at least one destination node of step **e** is accomplished by:

a. discovering links using the Intra Zone Routing Protocol;  
b. computing inbound trees for each of the destinations being queried;

- i. if any such trees exist and can be computed, then the nodes (other than the destination nodes) in these inbound trees denote the alternate destinations.

5        6. The method as set forth in claim 5 wherein the inbound trees are computed using  
a mechanism wherein a plurality of node neighbors having a link to the  
transmitting node are used to construct a graph of destinations; and for each  
destination, a shortest path protocol is executed on the graph by considering each  
destination as a sink node, any shortest path protocol can be used to compute  
10       these inbound trees.

7. The method as set forth in claim 6 wherein the shortest path protocol is Dijkstra's  
algorithm.

15       8. The method for routing data in wireless ad-hoc networks as set forth in claim 1,  
wherein the maximum number of hops between the first node and the border node  
is 3.

9. The method for routing data in wireless ad-hoc networks as set forth in claim 1,  
20       wherein the number of hops between the first node and the border node is  
periodically reset.

10. The method for routing data in wireless ad-hoc networks as set forth in claim 1,  
wherein the number of hops between the first node and the border node is  
dynamically updated based on a situational awareness protocol.
- 5 11. The method for routing data in wireless ad-hoc networks as set forth in claim 1,  
wherein there is a counter that limits the number of enhancements.
12. The method for routing data in wireless ad-hoc networks as set forth in claim 11,  
wherein the limit on the counter that limits the number of enhancements is  
10 determined by the method for routing data.
13. The method for routing data in wireless ad-hoc networks as set forth in claim 1,  
wherein there is a counter that limits the number of enhancements to a  
predetermined number is selected based on memory resources allocated to the  
15 method.
14. The method for routing data in wireless ad-hoc networks as set forth in claim 1,  
wherein the ENHANCEMENT\_INTERVAL is variable.
- 20 15. The method for routing data in wireless ad-hoc networks as set forth in claim 1,  
wherein the ENHANCEMENT\_INTERVAL is varied and the variation is based on the  
number of nodes adjacent to the source node.

16. The method for routing data in wireless ad-hoc networks as set forth in claim 1,  
wherein a transmission path is not the same as a return path.

17. The method for routing data in wireless ad-hoc networks as set forth in claim 1,

5 wherein the data packet includes at least one of the following:

- a. a set of node neighbors which have a link to the transmitting node;
- b. a set of node neighbors with which the transmitting node has a link;
- c. any nodes on the outbound tree of the transmitting node;
- d. a sequence number;
- 10 e. an urgent flag, which provides data concerning link status;
- f. a time to live counter which tracks the number of hops up to which the  
unit can be forwarded, the time to live counter is initialized to the  
ZONE\_RADIUS, and is incremented as the unit traverses a path.

15 18. The method for routing data in wireless ad-hoc networks as set forth in claim 17,  
wherein the nodes on the outbound tree of the transmitting node are computed  
from the units obtained from other nodes

19. An apparatus for routing data in wireless ad-hoc networks comprising:

20 a proactive element, configured to route messages utilizing an intra zone routing  
protocol; and



a reactive element, configured to route messages utilizing a inter zone routing protocol; and

a bordercast tree element, configured to bordercast to a plurality of border nodes; and

5

at least one query packet memory element comprising data, wherein nodes receiving one or more query packets are configured to provide at least one query response or discard the query packet;

10 wherein the apparatus includes:

i. a first means for initiating a route query from a first node or a source node and wherein the route query concerns a route to a destination node;

15 ii. a central means determines if there is a path to a destination node in an outbound tree, as computed by the proactive element, and, if such a path is found, that path is the desired path; and the apparatus transmits its message; otherwise;

20 iii. the source node checks if its bordercast tree is empty:

a. if the bordercast tree is empty go to step **viii**;

b. if the bordercast tree is not empty go to step **iv**;

iv. the bordercast tree is stored in the query packet, and is forwarded along the bordercast tree, one or more intermediate nodes of the bordercast tree (non- border nodes), forward the query packet until it reaches a border node, wherein a plurality of processing steps occur culminating in the sending of a bordercast:

a. after sending the bordercast, there is a pause for a predetermined period of time equal to `ENHANCEMENT_INTERVAL`, during which the source node awaits either a query response or one or more enhancement messages;

v. if a query response to the route query is received, then the route query step is termed complete and the computed route is returned to the first node;

vi. if a query response is not received, then the source node checks if an enhancement message has been received, the `ENHANCEMENT_INTERVAL` having passed since the initiation of the bordercast; if one or more query enhancement messages were received during the `ENHANCEMENT_INTERVAL`, then one or more alternate destination nodes suggested in the query enhancement message, or messages, are utilized to create an enhanced route query with an alternative set of destinations, wherein other nodes have reported that the alternative destination nodes have routes to the destination node; the new enhanced route query is processed like the original route query; by returning to step **ii**;

- vii. if the bordercasting did not result in any enhancement of the route query or in a route, the bordercast tree is incapable of reaching nodes that can enhance the query (assuming no message losses); this state is also reached from step **ii**, when the bordercast tree is empty; in this situation a two-way tree is used to send a request to enhance the query and the source node and the border nodes forward this Query Enhancement Request using the two-way tree just as they would forward a regular query, except that the two-way tree is used for bordercasting, instead of the bordercast tree; the objective here is to try to discover nodes, which know of paths to the destination node;
- viii. after waiting for `ENHANCEMENT_INTERVAL`, the source node checks to see if any responses to the query enhancement request using the two way tree exist; if one or more query enhancement responses are received during the `ENHANCEMENT_INTERVAL`, the resulting one or more destinations suggested in the query enhancement response can be queried for routes to the desired destination, if there is a path to a desired destination node in an outbound tree as computed by the proactive component then that path is the desired path and the protocol terminates;

ix. if there are any responses to the query enhancement request but there is not a path to the desired destination in an outbound tree as computed by the proactive component then go to step **iii**;

5 x. if no enhancement message was received then the destination is assumed to be unreachable and the protocol terminates.

20. The apparatus for routing data in wireless ad-hoc networks as set forth in claim

**19**, wherein in step **iv** the plurality of processing steps include:

- 10 a. receiving incoming query from step **iv** above;
- b. applying at least one query control mechanism to the query;
- c. using the criteria from the at least one Query Control Mechanism to drop unnecessary query threads,
- i. if all threads are dropped, then go to step **viii** of claim **19**;
- 15 d. determining if a path is known to at least one destination and
- i. if a path is known, notify sender of the path and route discovery is complete;
- ii. if a path is not known then go to the next step;
- e. checking if a path is known from other nodes to at least one destination
- 20 node:
- i. if yes, inform sender of alternative destinations;
- ii. if not;
- f. bordercasting query

g. returning to step **iv a** of claim **19**.

21. The apparatus as set forth in claim **20** wherein the at least one query control mechanism of step **b** includes a Query Detection (QD) and an Early Termination (ET) mechanism; these mechanisms are configured to extract a query identifier  
5 and match it with recently cached query identifiers seen by the node; if the query identifier has been seen before, then the query can be dropped.

22. The apparatus as set forth in claim **20** wherein the notification of a path to the sender, of step **d i** is accomplished by:

- 10
- a. initiating a response to the query that contains the computed path, only the border nodes, (also referred to as center nodes), traversed by the route query packet are recorded; the response is sent along a path that traverses the same center nodes; this is possible because each center node has a path  
15 to the previous center node, thus the response is forwarded from one center node to another center node until it reaches the source node, that initiated the query.

23. The apparatus as set forth in claim **20** wherein the checking if a path is known  
20 from other nodes to at least one destination node of step **e** is accomplished by:

- a. discovering links using the Intra Zone Routing Protocol;
- b. computing inbound trees for each of the destinations being queried;

- i. if any such trees exist and can be computed, then the nodes (other than the destination nodes) in these inbound trees denote the alternate destinations.

5        24. The apparatus as set forth in claim **23** wherein the inbound trees are computed using a mechanism wherein a plurality of node neighbors having a link to the transmitting node are used to construct a graph of destinations; and for each destination, a shortest path protocol is executed on the graph by considering each destination as a sink node, any shortest path protocol can be used to compute  
10        these inbound trees.

25. The apparatus as set forth in claim **24** wherein the shortest path protocol is Dijkstra's algorithm.

15        26. The apparatus for routing data in wireless ad-hoc networks as set forth in claim **19**, wherein the maximum number of hops between the first node and the border node is 3.

27. The apparatus for routing data in wireless ad-hoc networks as set forth in claim  
20        **19**, wherein the maximum number of hops between the first node and the border node is periodically reset.

28. The apparatus for routing data in wireless ad-hoc networks as set forth in claim 19, wherein the maximum number of hops between the first node and the border node is dynamically updated based on a situational awareness protocol.

5 29. The apparatus for routing data in wireless ad-hoc networks as set forth in claim 19, wherein there is a limiting counter configured to terminate the protocol after a predetermined number of enhancements.

10 30. The apparatus for routing data in wireless ad-hoc networks as set forth in claim 29, wherein the counter that limits the number of enhancements is determined by the method for routing data.

15 31. The apparatus for routing data in wireless ad-hoc networks as set forth in claim 29, wherein there is a counter that limits the number of enhancements to a predetermined number is selected based on memory resources allocated to the method.

20 32. The apparatus for routing data in wireless ad-hoc networks as set forth in claim 19, wherein the ENHANCEMENT\_INTERVAL is varied.

33. The apparatus for routing data in wireless ad-hoc networks as set forth in claim 19, wherein the ENHANCEMENT\_INTERVAL is varied and the variation is based on the number of nodes adjacent to the source node.

34. The apparatus for routing data in wireless ad-hoc networks as set forth in claim  
**19**, wherein a transmission path is not the same as a return path.

5 35. The apparatus for routing data in wireless ad-hoc networks as set forth in claim  
**19**, wherein the data packet includes at least one of the following:

- a. a set of node neighbors which have a link to the transmitting node;
- b. a set of node neighbors with which the transmitting node has a link;
- c. any nodes on the outbound tree of the transmitting node;
- 10 d. a sequence number;
- e. an urgent flag, which provides data concerning link status;
- f. a time to live counter which tracks the number of hops up to which the  
unit can be forwarded, the time to live counter is initialized to the  
ZONE\_RADIUS, and is incremented as the unit traverses a path.

15 36. The apparatus for routing data in wireless ad-hoc networks as set forth in claim  
**35**, wherein the nodes on the outbound tree of the transmitting node are computed  
from the units obtained from other nodes.

20 37. An apparatus for routing data in a wireless ad-hoc network when unidirectional  
links are present including an on-demand search mechanism configured to  
recursively attempt to build a path to a destination node by identifying and  
utilizing nodes having a route to the destination.